编译技术与编译教学新的发展契机

计卫星

2019年11月22日

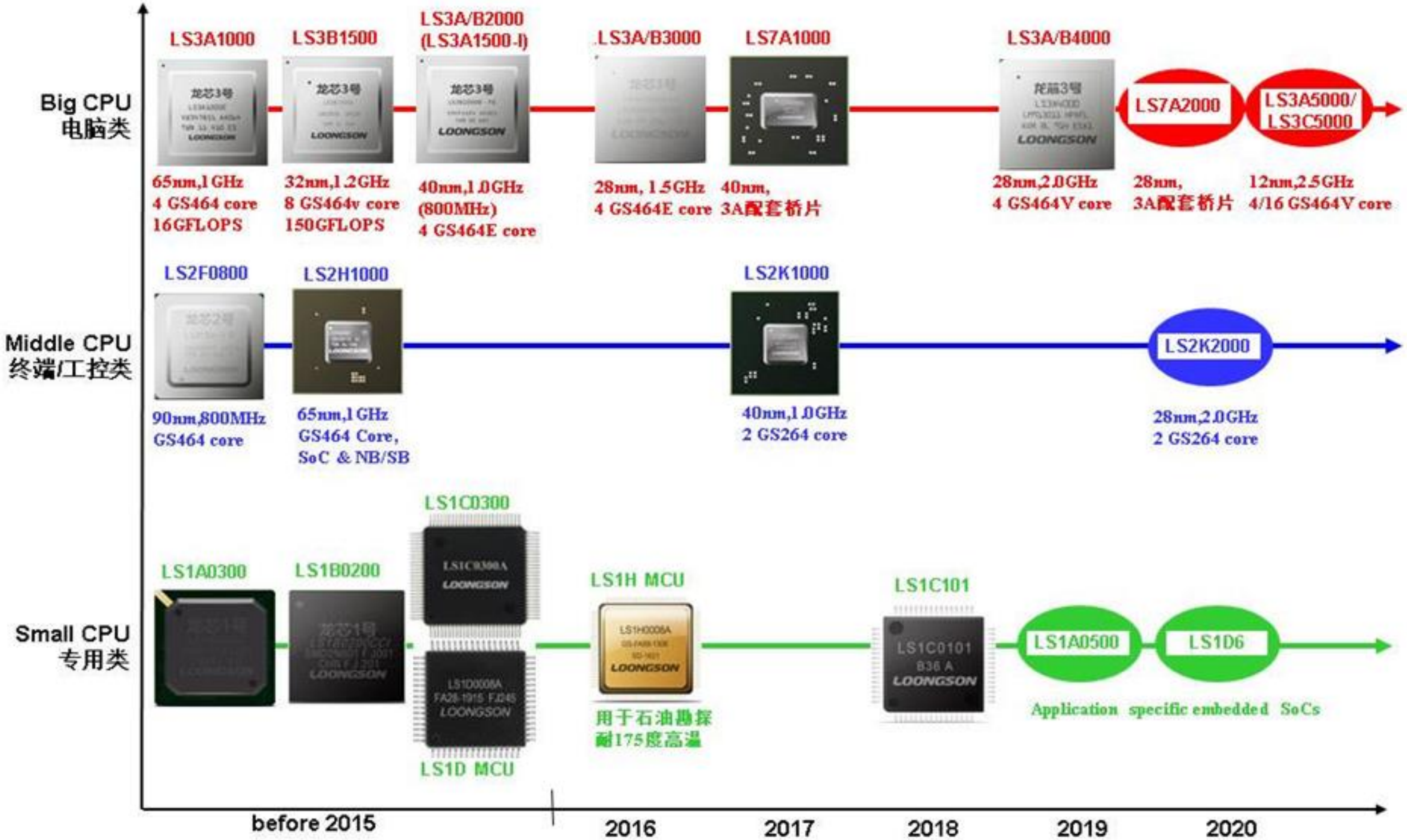
北京理工大学计算机学院
School of Computer Science and Technology,BIT

内容提纲

- 国产处理器芯片加速发展
- 华为方舟编译器初露头角
- GraalVM逐步成熟
- 关于编译教学的探究
- 总结

国产处理器芯片加速发展

- 龙芯



国产处理器芯片加速发展

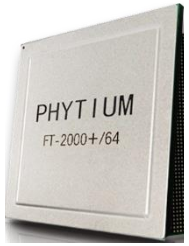
- 其他

兆芯

MPRC
北大众志

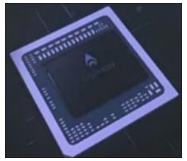
申威

PHYTIUM 飞腾



国产处理器芯片加速发展

- 比特大陆



2013年
公司创立

2013年底
发布首款区块链芯片

2014年
多款产品畅销市场

2015年
区块链市场领先
启动AI业务

2016年
S9 矿机市场销量第一
第一代AI芯片下线

2017年
人工智能产品面世
营收利润大幅增长

2018年
第二代AI产品问世

2019年
第三代AI产品问世

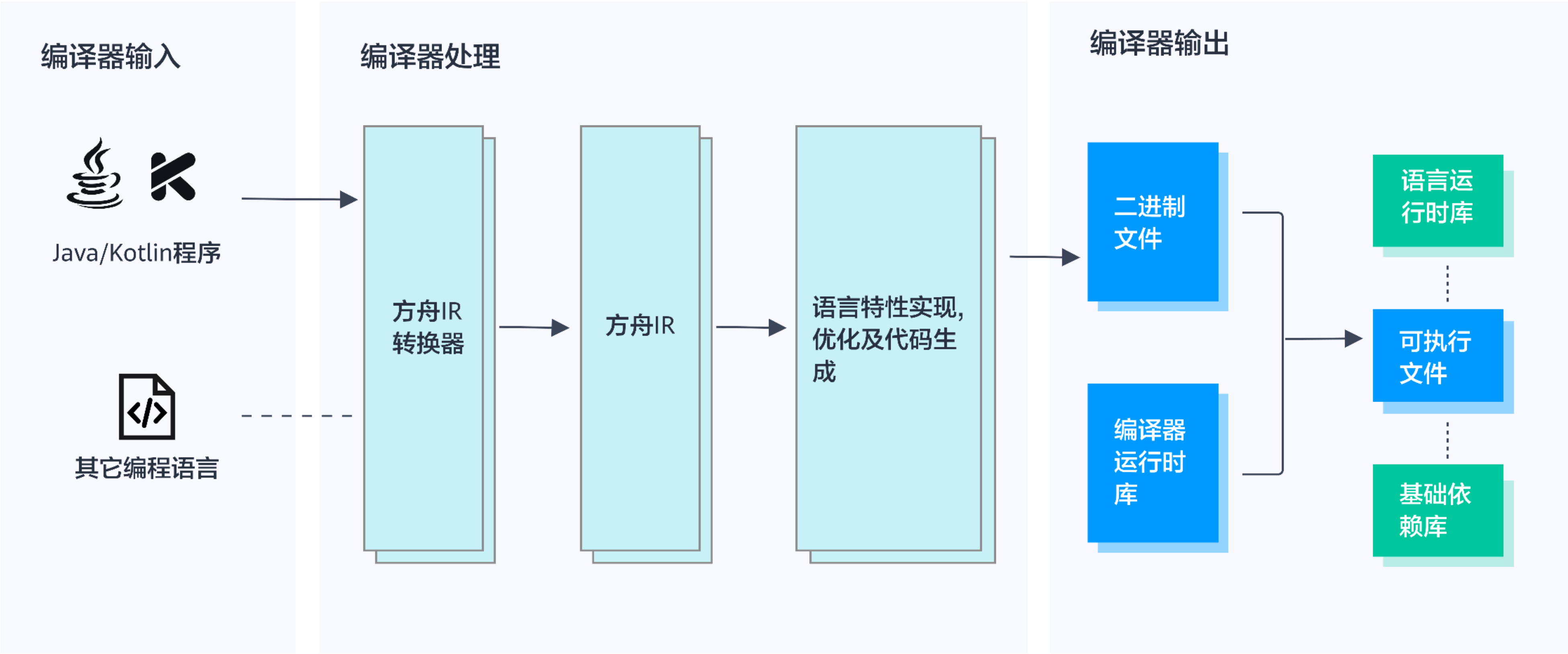
2020年
第四代AI产品

Sophon BM1686

国产处理器芯片加速发展

- 2016：清微智能-Thinker 1
- 2016：寒武纪1A
- 2017：地平线视觉芯片征程、旭日
- 2018：寒武纪云端MLU100
- 2019：地平线-征程二代
- 2019：平头哥-玄铁910
- 2019：平头哥-含光800AI
- 2019：清华-天机芯
- 2019：清微智能-多模态TX510

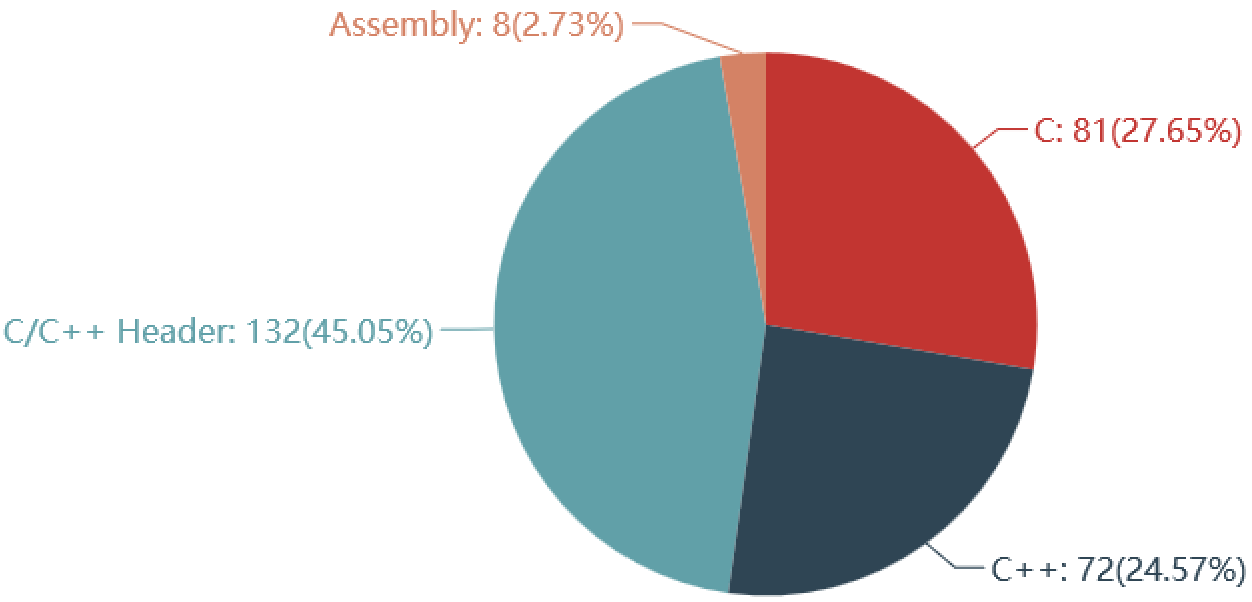
华为方舟编译器初露头角



V

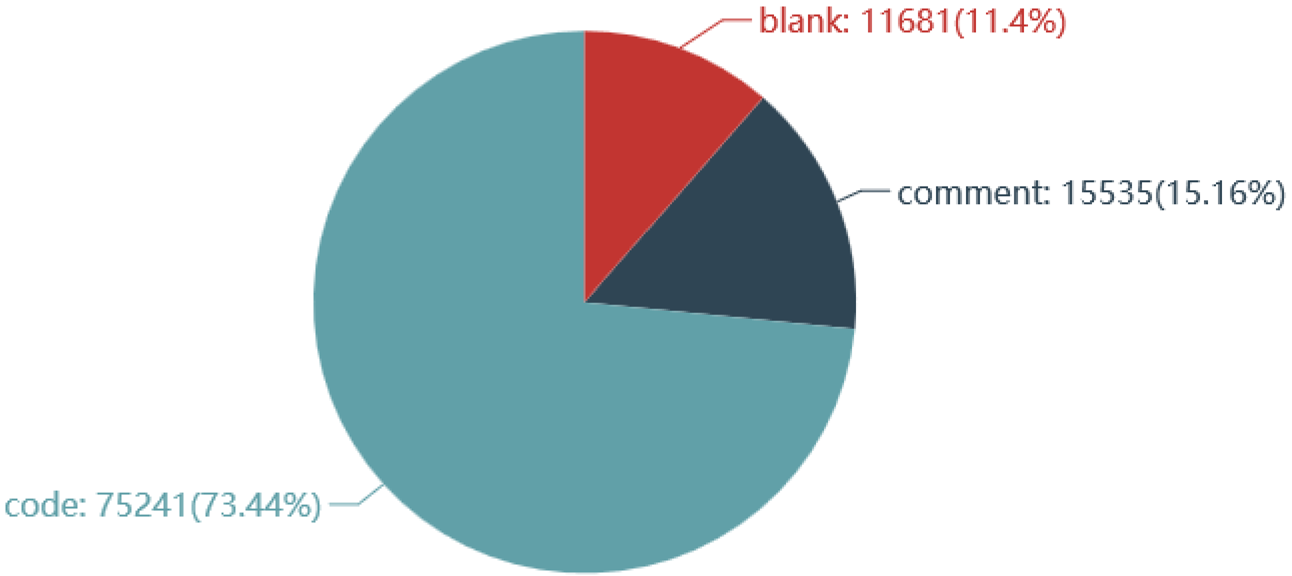
华为方舟编译器初露头角

- 项目源码规模



华为方舟编译器初露头角

- 项目源码规模



华为方舟编译器初露头角

- 测试结果

序号	行数	结果	序号	行数	结果
1	11	√	6	10	√
2	31	√	7	24	√
3	24	√	8	7	√
4	44	√	9	19	√
5	73	√	10	17	√

序号	文件个数	行数
1	22	5, 059
2	83	52, 468

华为方舟编译器初露头角

- 中间代码设计-MAPLE
  - 尽可能保留源码信息
  - 高层次树状层次化结构
  - 低层次与指令一一对应
  - 可扩展-支持新的语言和控制结构

华为方舟编译器初露头角

- 相关对比：MAPLE & JBC

```
int fact(int n){
  if( n!=1 )
      return n * fact(n -1);
  else
      return 1;
```

```
int &fact(var %n i32) i32 {
  if(ne i32 (dread i32 %n, constval i32 1)){
      call $fact(sub i32 (dread i32 %n, constval i32 1))
      return (mul i32 (dread i32 %n, regread i32 %%retval))
  }
  return(constval i32 1)
```

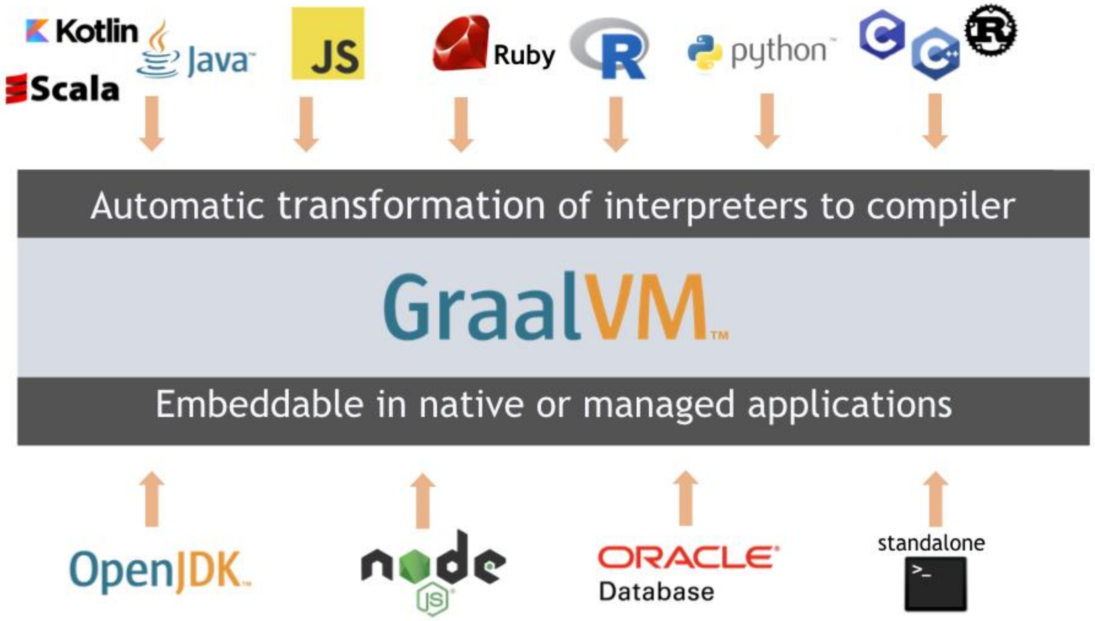
```
 int fact(int);
    Code:
        0: iload_1
        1: iconst_1
        2: if_icmpeq      15
        5: iload_1
        6: aload_0
        7: iload_1
        8: iconst_1
        9: isub
       10: invokevirtual #5
// Method fact:(I)I
       13: imul
       14: ireturn
       15: iconst_1
       16: ireturn
```

GraalVM逐步成熟

- 主要做法
  - 抽取语言共性，建立中间层API
- 主要原因
  - “That library is not available in my language. I need to rewrite it.”
  - “That language would be the perfect fit for my problem, but we cannot run it in our environment.”
  - “That problem is already solved in my language, but the language is too slow.”

GraalVM逐步成熟

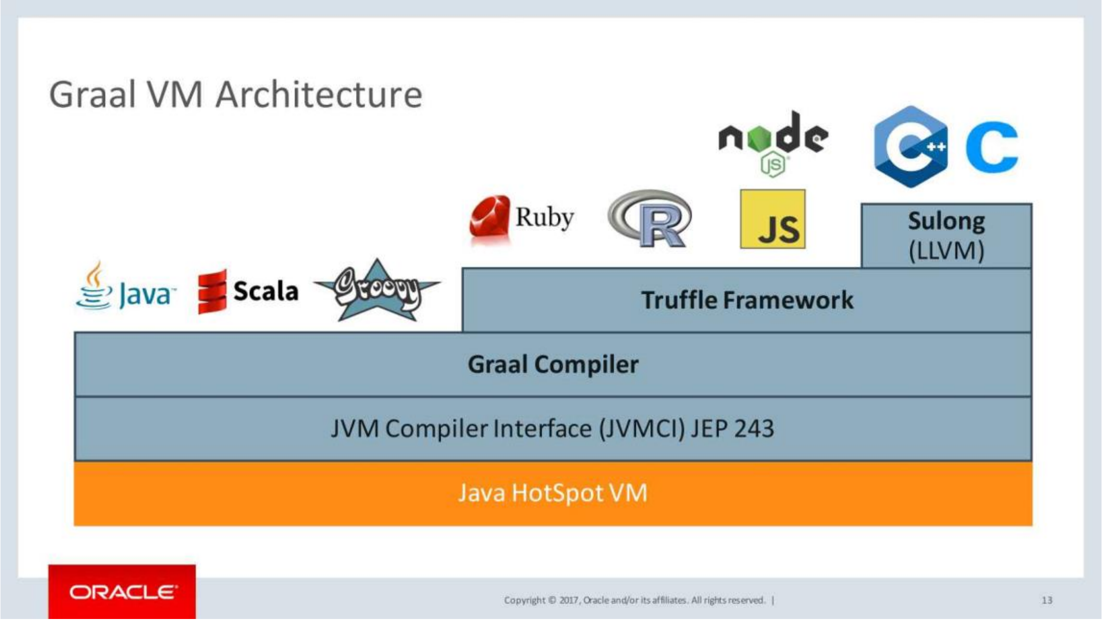
- 多语言互操作




```
const express = require('express');
const app = express();
app.listen(3000);
app.get('/', function(req, res) {
  var text = 'Hello World!';
  const BigInteger = Java.type(
    'java.math.BigInteger');
  text += BigInteger.valueOf(2)
    .pow(100).toString(16);
  text += Polyglot.eval(
    'R', 'runif(100)')[0];
  res.send(text);
})
```

GraalVM逐步成熟

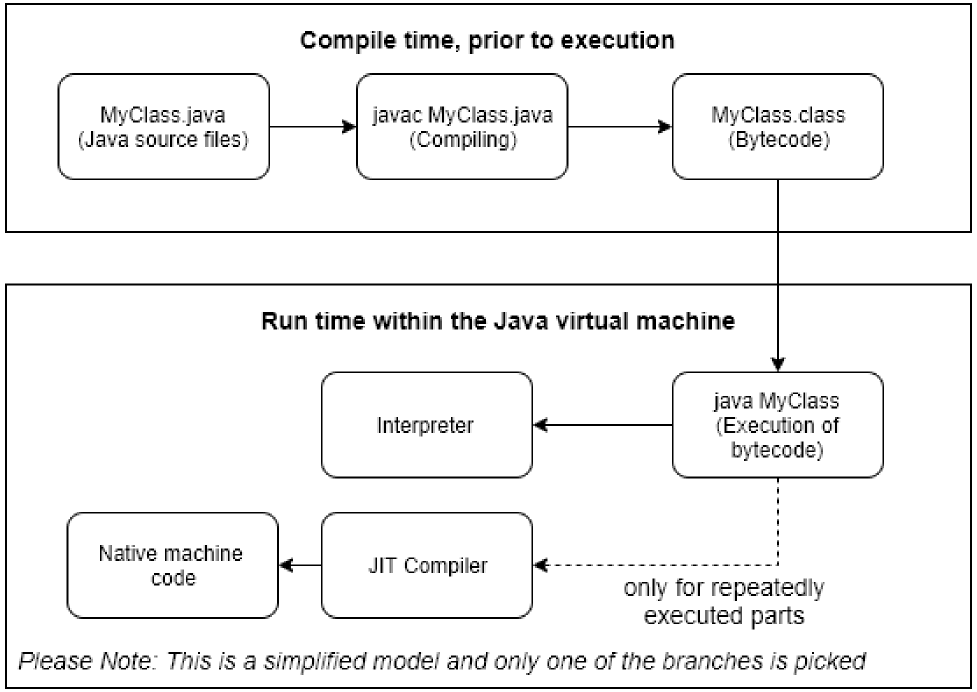
- Polyglot

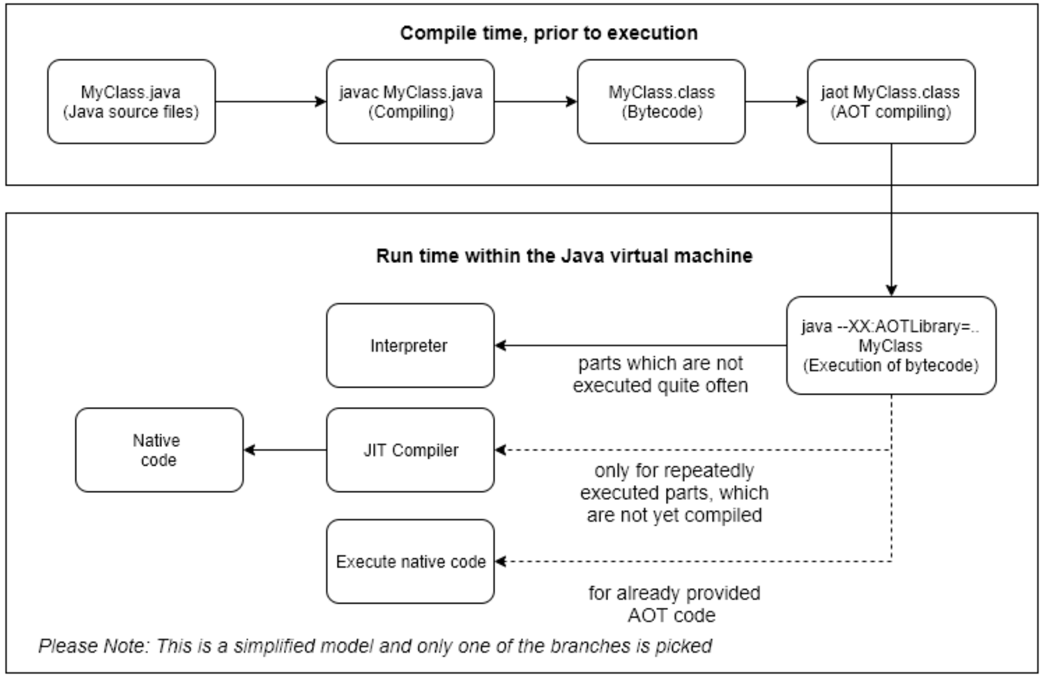



```java
import org.graalvm.polyglot.*;
public class HelloPolyglot {
  public static void main(String[] args) {
    System.out.println("Hello Java!");
    Context context = Context.create();
    context.eval("js",
      "print('Hello JavaScript!');");
  }
}
```

GraalVM逐步成熟

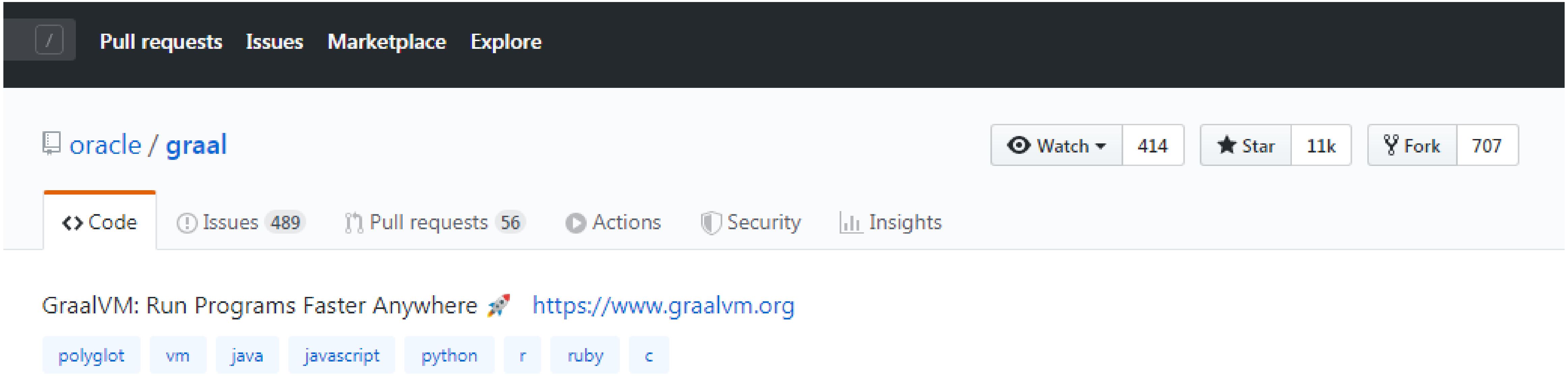
- JVM和GraalVM对比





https://rieckpil.de/whatis-graalvm/

GraalVM逐步成熟



GraalVM逐步成熟

- 代码提交情况

大学计算机学院
Science and Technology,BIT

GraalVM逐步成熟

- GraalVM项目负责人

Thomas Wuerthinger

Oracle Labs
Verified email at oracle.com - Homepage
Computer Science

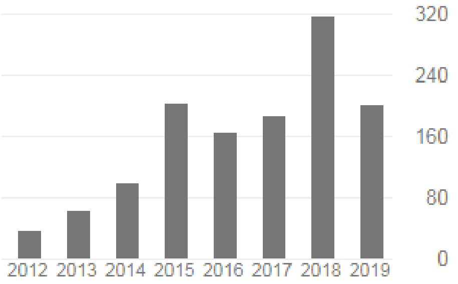
FOLLOW

GET MY OWN PROFILE

TITLE	CITED BY	YEAR
One VM to rule them all T Würthinger, C Wimmer, A Wöß, L Stadler, G Duboscq, C Humer, ... Proceedings of the 2013 ACM international symposium on New ideas, new ...	213	2013
Self-optimizing AST interpreters T Würthinger, A Wöß, L Stadler, G Duboscq, D Simon, C Wimmer Acm Sigplan Notices 48 (2), 73-82	127	2012
Dynamic code evolution for Java T Würthinger, C Wimmer, L Stadler Proceedings of the 8th International Conference on the Principles and ...	82	2010
An intermediate representation for speculative optimizations in a dynamic compiler G Duboscq, T Würthinger, L Stadler, C Wimmer, D Simon, H Mössenböck	81	2013

Cited by VIEW ALL

	All	Since 2014
Citations	1387	1174
h-index	19	18
i10-index	33	29

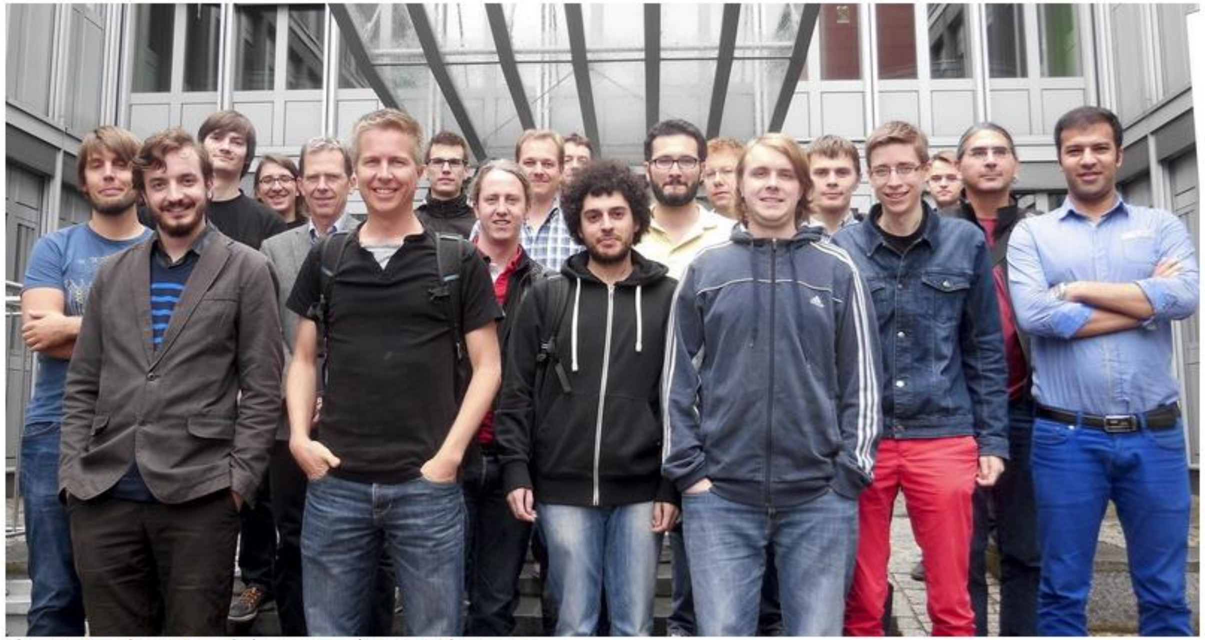


GraalVM逐步成熟

- 林茨大学

Team

Oracle Labs @ JKU		PhD students	Graduate students	Former researchers	
Thomas Würthinger (lead)	Peter Hofer	Benoit Daloze	Josef Haider	Thomas Feichtinger	Thomas Kotzmann
Florian Angerer	Roland Schatz	Josef Eisl	Jacob Kreindl	Gilles Duboscq	Michael Pfeiffer
Danilo Ansaloni	Thomas Schatzl	David Leopoldseder	Raphael Mosaner	Miguel Garcia	Stefan Rumzucker
Stefan Anzinger	Lukas Stadler	Manuel Rigger	Daniel Pekarek	Reem Hourieh	Christoph Schmid
Daniele Bonetta	Christian Wirth		Thomas Pointhuber	Christian Humer	Bernhard Urban
Matthias Grimmer	Andreas Wöß	Hanspeter Mössenböck (supervisor)		Christian Huber	Christian Wimmer
Christian Häubl				Christos Kotselidis	



The team at the JVM workshop at ETH in September 2014

关于编译教学的探究

- 个人经历
  - GCC/LLVM
  - Determinism Parallel Ruby
  - 基于QEMU获取trace
  - 代码分析
  - 算法优化
  - GraalVM
  - BIT-MiniCC/MiniCCompiler
  - …

关于编译教学的探究

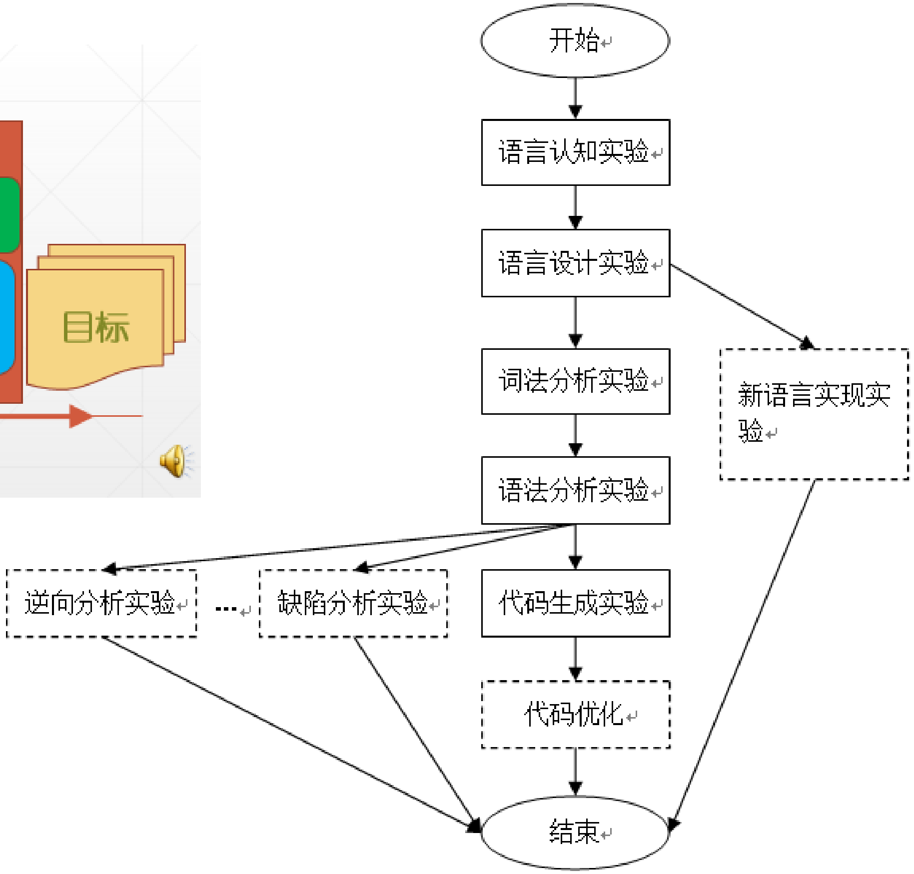
- 主要的问题
  - 玩具型与工业实用型？
  - 传统语言与新语言？
  - 国外开源与国内自主可控？
- 为什么他们可以
  - GCC/LLVM
  - RISC-V + Chisel
  - GraalVM
  - Pin-tool
  - ANTLR
  - …

关于编译教学的探究

- 理想型实验
  - 模块可替换
  - 难度阶梯型
  - 10万行代码规模/近工业强度
  - 系统兼容性强
  - 对接到开源社区
  - 鼓励新语言探索
  - …

关于编译教学的探究

- 现有做法是否合适？



关于编译教学的探究

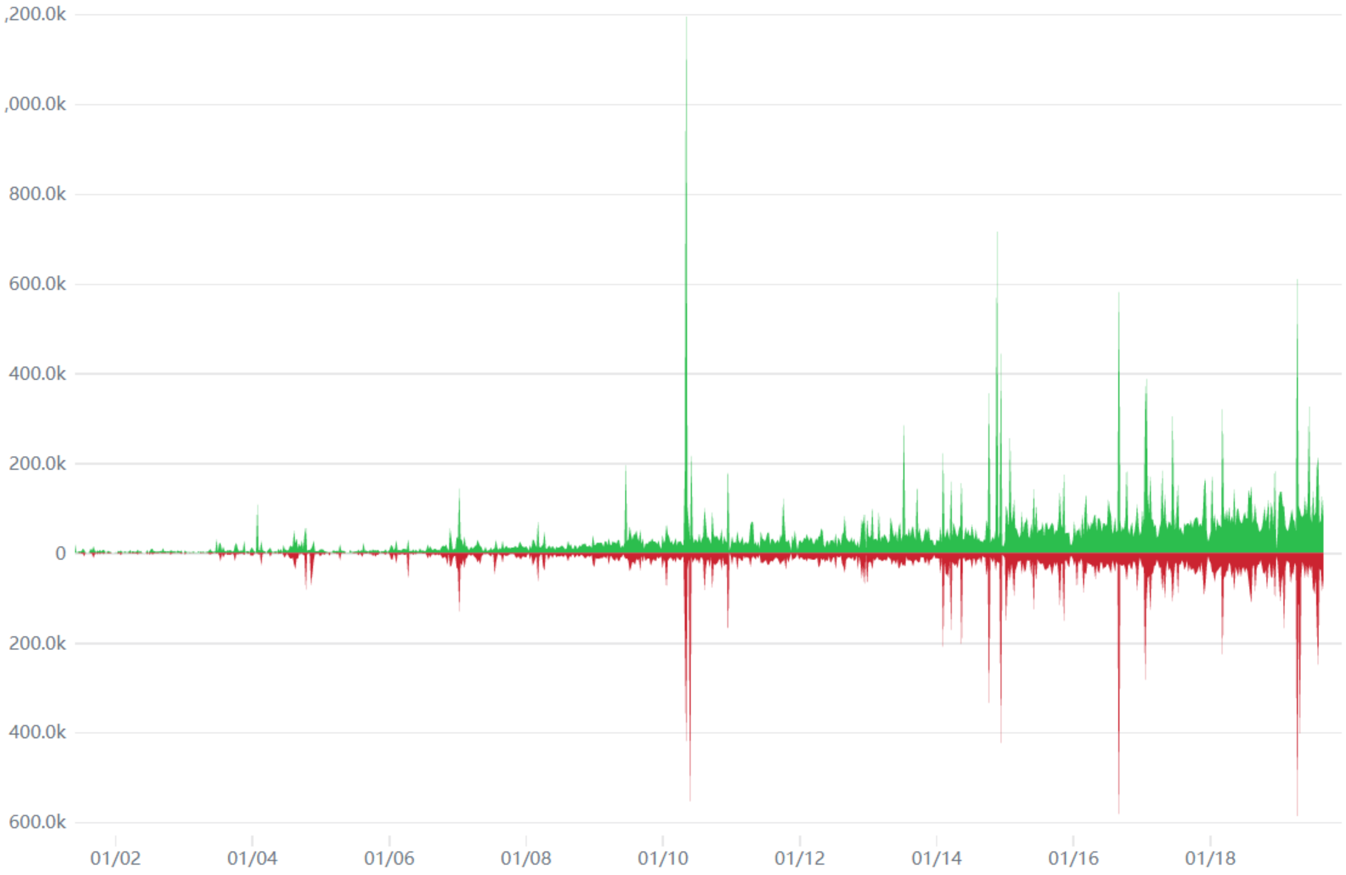
- 更好的建设途径
  - 专项支持：申请项目建设
  - 产学合作：企业授权，案例设计与构造
  - 开源改造：持续改造
  - 竞赛积累：自我生长，持续改进
  - 教研相长：特色鲜明，单点突破
  - …

关于编译教学的探究

- 相关领域的经验
  - 系统能力（组成和体系结构）
    - 广泛参与，持续改进，成绩不断被推高
  - 并行计算
    - 国外厂商和国内厂商不同竞赛
  - 软件工程
    - 原型系统竞赛
  - 操作系统
  - 编译原理

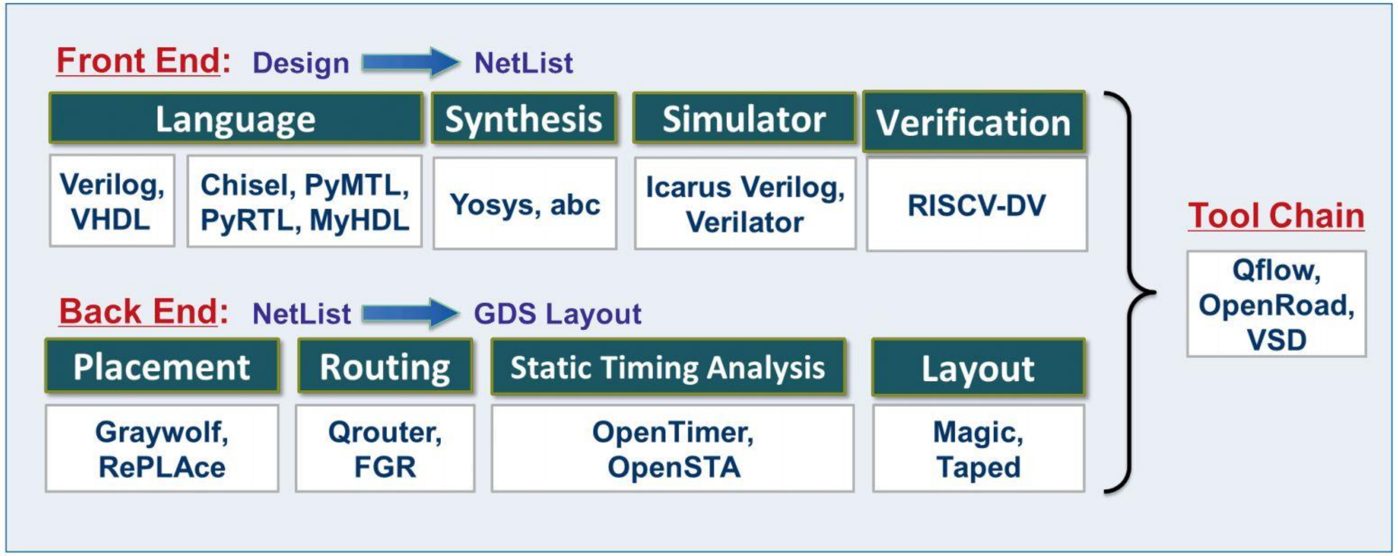
关于编译教学的探究

- LLVM代码提交情况



关于编译教学的探究

- 新的机遇
  - 大数据近似计算
  - 硬件加速器
  - 量子计算
  - EDA工具链
  - 开源芯片
  - …



http://blog.sciencenet.cn/home.php?mod=space&uid=414166&do=blog&id=1203989

总 结

- 国产芯片呈现百花盛开的局面
- 提升编译人才数量和质量需共同发力
- 编译技术
  - 超前布局、单点突破、超越式发展
- 编译教学
  - 产学合作、长期坚持、重视开源生态对接